



Yang, R., Yang, C., Chen, M., & Na, J. (2016). RBFNN based adaptive control of uncertain robot manipulators in discrete time. In *2016 UKACC 11th International Conference on Control (CONTROL 2016): Proceedings of a meeting held 31 August - 2 September 2016, Belfast, United Kingdom* [7737617] Institute of Electrical and Electronics Engineers (IEEE).
<https://doi.org/10.1109/CONTROL.2016.7737617>

Peer reviewed version

Link to published version (if available):
[10.1109/CONTROL.2016.7737617](https://doi.org/10.1109/CONTROL.2016.7737617)

[Link to publication record in Explore Bristol Research](#)
PDF-document

This is the author accepted manuscript (AAM). The final published version (version of record) is available online via IEEE at <http://ieeexplore.ieee.org/document/7737617/>. Please refer to any applicable terms of use of the publisher.

University of Bristol - Explore Bristol Research

General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

RBFNN Based Adaptive Control of Uncertain Robot Manipulators in Discrete Time

Runxian Yang^{1,2,3}, Chenguang Yang^{2*}, Mou Chen¹ and Jing Na⁴

¹ College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China

² Zienkiewicz Centre for Computational Engineering, Swansea University, Swansea, UK

³ College of Electric and IT, Yangzhou Polytechnic Institute, Yangzhou, China

⁴ Department of Mechanical Engineering, University of Bristol, Bristol, UK

Abstract—The trajectory tracking control problem for a class of n -degree-of-freedom (n -DOF) rigid robot manipulators is studied in this paper. A novel adaptive radial basis function neural network (RBFNN) control is proposed in discrete time for multiple-input multiple-output (MIMO) robot manipulators with nonlinearity and time-varying uncertainty. The high order discrete-time robot model is transformed to facilitate digital implementation of controller, and the output-feedback form is derived to avoid potential noncausal problem in discrete time. Furthermore, the desired controller based on RBFNN is designed to compensate for effect of uncertainties, and the RBFNN is trained using tracking error, such that the stability of closed-loop robot system has been well guaranteed, the high-quality control performance has been well satisfied. The RBFNN weight adaptive law is designed and the semi-global uniformly ultimate boundedness (SGUUB) is achieved by Lyapunov based on control synthesis. Comparative simulation studies show the proposed control scheme results in supreme performance than conventional control methods.

I. INTRODUCTION

WITH advances of technologies, robot applications in industry and our daily life become increasingly popular, the relevant research works have been an attractive topic. However, most robot manipulators are usually subject to unmodelled dynamics and various uncertainties in practice [2]–[4], an ideal control design for a class of robot manipulators is challenging.

Various approaches for trajectory tracking control of robot manipulators have been proposed. Feedback linearization methods [5], [6], sliding mode and other robust control methods [1], [7]–[9] have all been extensively investigated for robot control, and global tracking error convergence are able to be guaranteed. Furthermore, the advanced intelligent methods and relevant research results have been well applied to robot control, e.g., adaptive control [10], [11], adaptive-fuzzy control [12], adaptive-sliding control [13], and complex adaptive control based on fuzzy and sliding-mode theories for robot manipulators [14], function approximators have also been utilized. In order to compensate for uncertainties of

robot manipulators, adaptive neural network (ANN) techniques have been popular in recent years [15]–[17], ANNs have universal approximation capability for nonlinear functions. An adaptive RBFNN algorithm guaranteeing closed-loop stability has been proposed for robot manipulator systems in [18]. A novel RBFNN estimator has been designed to compensate for uncertainties in [19], [20]. These approaches are able to guarantee UUB of closed-loop system of robot manipulators. But the digital implementation of robot controllers and network communication of high-speed computers are becoming increasingly popular and powerful. Thus, the increasing research works for robot manipulators have now been carried out in discrete time.

Discrete-time robot manipulator models and discrete-time control methods are used in [18], [21], the discrete-time controllers applied to on-line robot control provides convenience for implementation. In [22], a ANN controller has been proposed based on combining one-step-ahead control with ANN control for a class of MIMO discrete-time systems with nonaffine nonlinearity. In [28], a class of MIMO nonlinear systems with block triangular structure can be decomposed in discrete time, by applying pure-feedback method, an ANN control has been presented based on all subsystems with couplings and unknown directions. These discrete-time approaches perform well to guarantee robust stability of nonlinear robot systems. However, these research works only guarantee stability of closed-loop robot manipulator systems, while realizing trajectory tracking control is seldom in discrete time. Thus, a novel control scheme proposed for a class of robot manipulators with uncertainty in discrete time is the main research objectives of this paper.

Aiming to address satisfied trajectory tracking performance based on stable closed-loop robot system, we develop a discrete-time novel RBFNN based adaptive control for uncertain robot manipulators.

The following notations are employed in this paper.

- $\|\cdot\|$ represents the Euclidean norm of vectors and induced norm of matrices.
- $b := a$ denotes b is defined as a .
- $[\]^T$ represents the transpose of a vector or a matrix.
- $[\]^{-1}$ represents the inverse of a n -order reversible matrix.

This work was supported in part by Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/L026856/2, and Marie Curie Intra-European Fellowships Project AECE under Grant FP7-PEOPLE-2013-IEF-62553174.

*Corresponding author. Email: cyang@theiet.org

- $\mathbf{0}_{[p]}$ denotes the dimension of zero vector is p -dimension.
- $\mathbf{I}_{[m]}$ stands for m -dimension unit matrix.
- W^* represents the idea neural net weight matrix.
- \hat{W}^k represents the estimate value matrix of neural net idea weight W^* at the k -th step.
- $\tilde{W}^k = \hat{W}^k - W^*$ denotes the weight estimate error.

II. DISCRETIZING FOR ROBOT MODEL

The dynamic model of general n -DOF nonlinear rigid robot manipulators can be described using ordinary differential equation as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau + \tau_d \quad (1)$$

where $q \in \mathbb{R}^n$ is the joint position, and $\dot{q} \in \mathbb{R}^n$ is the joint velocity, $\ddot{q} \in \mathbb{R}^n$ is the joint acceleration, $M(q) \in \mathbb{R}^{n \times n}$ is the symmetric and positive definite inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ is the Coriolis-Centrifugal torque matrix, $G(q) \in \mathbb{R}^n$ denotes the gravity torque vector, $\tau \in \mathbb{R}^n$ is the control input torque vector, $\tau_d \in \mathbb{R}^n$ is the external force torque vector.

According to [2], the following properties hold for rigid robot manipulators in (1):

Property 1: $M(q)$ is uniformly bounded, and satisfies the following inequality

$$\underline{m} \leq \|M(q)\| \leq \bar{m} \quad (2)$$

Property 2: The matrix $C(q, \dot{q})$ and the vector $G(q)$ are bounded by $\|C(q, \dot{q})\| \leq k_c \|\dot{q}\|$, and $\|G(q)\| \leq k_g$, respectively, where k_c and k_g are positive constants.

It is very important and meaningful to design robot controller in discrete time. For a class of n -DOF rigid nonlinear robot manipulators with uncertainty in (1), which can be discretized by using discretization theory with a small sampling time interval T . The sampled joint angle is $q^k = q^{t_k}$, the sampled joint angle velocity is $\dot{q}^k = \dot{q}^{t_k}$, the control torque is $\tau^k = \tau^{t_k}$ and the external disturbance torque is $\tau_d^k = \tau_d^{t_k}$ at the sampling time instant $t_k = kT$, respectively.

Define $p^k = q^k \in \mathbb{R}^n$ and $v^k = \dot{q}^k \in \mathbb{R}^n$, then, the equivalent dynamics form in discrete time can be obtained [23], [26], [27] as

$$(M(\xi^k)/T)(v^{k+1} - v^k) = (M(\xi^k) - M(p^k))v^k - f(p^k, v^k) + \tau^k + \tau_d^k \quad (3)$$

where $M(\xi^k) \in \mathbb{R}^{n \times n}$ is also the inertia matrix with $\xi^k \cong p^k + Tv^k \in \mathbb{R}^n$, $f(p^k, v^k) = C(p^k, v^k)v^k + G(p^k) \in \mathbb{R}^n$, $C(p^k, v^k) \in \mathbb{R}^{n \times n}$ is Coriolis-Centrifugal torque matrix and $G(p^k) \in \mathbb{R}^n$ is gravitational synthetic torque vector in discrete time, respectively.

According to Property 1, $M(\xi^k)$ is also symmetric, positive definite and bounded, satisfying $\underline{m} \leq \|M(\xi^k)\| \leq \bar{m}$ with known constants $\underline{m} > 0$ and $\bar{m} > 0$.

III. TRANSFERING TO FEEDBACK SYSTEM

To avoid possible noncausal problem in control design, we extend our previous research works [24], [29], [30] to a class of nonlinear time-varying MIMO robot manipulators with uncertainty in discrete time.

The discrete-time dynamics in (3) can be transferred into the output-feedback control system [31] as

$$\begin{cases} p^{k+1} = p^k + Tv^k \\ v^{k+1} = [(1+T)\mathbf{I}_{[n]} - TM^{-1}(\xi^k)M(p^k) \\ \quad - TM^{-1}(\xi^k)C(p^k, v^k)]v^k - TM^{-1}(\xi^k)G(p^k) \\ \quad + TM^{-1}(\xi^k)\tau^k + TM^{-1}(\xi^k)\tau_d^k \end{cases} \quad (4)$$

where τ_d^k is bounded as $\|\tau_d^k\| \leq \bar{\tau}_d$ with an known constant $\bar{\tau}_d$

It is easily known that $M^{-1}(\xi^k)$ is also bounded, satisfying $\underline{m}^* \leq \|M^{-1}(\xi^k)\| \leq \bar{m}^*$ with known constants $\underline{m}^* > 0$ and $\bar{m}^* > 0$.

The control objective is to synthesize an adaptive RBFNN control input τ^k for robot system (4), not only all signals of closed-loop robot system are bounded, but also the joint position signal p^k is able to well track the ideal trajectory signal of robot manipulators $p_d^k \in \Omega_{p_d}^k$, finally, the satisfied control performance is able to be obtained, where Ω_{p_d} is a compact set.

It is noted that v^{k+1} depends on control output τ^k , while p^{k+1} is associated with p^k and v^k at the $(k+1)$ -th step in (4).

We can rewrite the first equation of the system (4) as $p^{k+1} - p^k - Tv^k = \mathbf{0}_{[n]}$, and v^k is designed as $v^k = \frac{1}{T}(p^{k+1} - p^k)$. To predict the $(k+2)$ th step of robot manipulators, we have

$$\begin{aligned} p^{k+2} &= p^{k+1} + Tv^{k+1} \\ &= [(2+T)\mathbf{I}_{[n]} - TM^{-1}(\xi^k)M(p^k) \\ &\quad - TM^{-1}(\xi^k)C(p^k, v^k)]p^{k+1} \\ &\quad - [(1+T)\mathbf{I}_{[n]} - TM^{-1}(\xi^k)M(p^k) \\ &\quad - TM^{-1}(\xi^k)C(p^k, v^k)]p^k \\ &\quad - T^2M^{-1}(\xi^k)G(p^k) \\ &\quad + T^2M^{-1}(\xi^k)\tau^k + T^2M^{-1}(\xi^k)\tau_d^k \end{aligned} \quad (5)$$

Furthermore, we need to move (5) back to the $(k+1)$ -th step, the output-feedback method is applied to get the p^{k+1} as

$$\begin{aligned} p^{k+1} &= [(2+T)\mathbf{I}_{[n]} - TM^{-1}(\xi^{k-1})M(p^{k-1}) \\ &\quad - TM^{-1}(\xi^{k-1})C(p^{k-1}, v^{k-1})]p^k \\ &\quad - [(1+T)\mathbf{I}_{[n]} - TM^{-1}(\xi^{k-1})M(p^{k-1}) \\ &\quad - TM^{-1}(\xi^{k-1})C(p^{k-1}, v^{k-1})]p^{k-1} \\ &\quad - T^2M^{-1}(\xi^{k-1})G(p^{k-1}) \\ &\quad + T^2M^{-1}(\xi^{k-1})\tau^{k-1} + T^2M^{-1}(\xi^{k-1})\tau_d^{k-1} \end{aligned} \quad (6)$$

Substituting (6) to (5), we note that there is no more explicit future outputs and input signals. For convenience, let us define

$$\begin{aligned} L^k &= (2+T)\mathbf{I}_{[n]} - TM^{-1}(\xi^k)M(p^k) - TM^{-1}(\xi^k)C(p^k, v^k) \\ R^k &= (1+T)\mathbf{I}_{[n]} - TM^{-1}(\xi^k)M(p^k) - TM^{-1}(\xi^k)C(p^k, v^k) \\ M_\tau^k &= T^2M^{-1}(\xi^k), \quad G^k = G(p^k) \end{aligned}$$

Considering equation (6), we know that future state at the $(k+1)$ -th step is able to be obtained by getting values of the

current k -th step and the past $(k-1)$ -th step. Then, the output p^{k+2} is obtained as

$$\begin{aligned} p^{k+2} = & (L^k L^{k-1} - R^k) p^k - L^k R^{k-1} p^{k-1} \\ & - L^k M_\tau^{k-1} G^{k-1} - M_\tau^k G^k + L^k M_\tau^{k-1} \tau^{k-1} \\ & + M_\tau^k \tau^k + L^k M_\tau^{k-1} \tau_d^{k-1} + M_\tau^k \tau_d^k \end{aligned} \quad (7)$$

and we further define

$$\begin{aligned} L_p^k &= (L^k L^{k-1} - R^k) p^k - L^k R^{k-1} p^{k-1} + L^k M_\tau^{k-1} \tau^{k-1} \\ L_G^k &= L^k M_\tau^{k-1} G^{k-1} + M_\tau^k G^k \\ L_d^k &= L^k M_\tau^{k-1} \tau_d^{k-1} + M_\tau^k \tau_d^k \end{aligned}$$

Thus, equation (7) can be rewritten as

$$\begin{aligned} p^{k+2} &= L_p^k - L_G^k + M_\tau^k \tau^k + L_d^k \\ &= \psi(p^{k-1}, p^k, \tau^k, \tau^{k-1}, \tau_d^{k-1}, \tau_d^k) \end{aligned} \quad (8)$$

It is easily known that the function $\psi(\cdot, \cdot, \cdot, \cdot, 0, 0)$ in (8) is continuous for all the arguments and continuously differentiable.

Lemma 1: M_τ^k is symmetric positive definite matrix, and is bounded as $\underline{m}_\tau \leq M_\tau^k \leq \bar{m}_\tau$ with $\underline{m}_\tau = T^2 \underline{m}^*$ and $\bar{m}_\tau = T^2 \bar{m}^*$.

According to Lemma 1, we know that L_d^k is bounded and $\|L_d^k\| \leq (3 + 2T + T\bar{m}^* k_c) \bar{m}_\tau \bar{\tau}_d := \bar{\tau}_d^*$.

IV. ADAPTIVE RBFNN CONTROLLER DESIGN

A. RBFNN Approximation

The RBFNN can approximate any nonlinear function $F(z)$, which can be expressed as [25]:

$$F(z) = W^T S(z), \quad W \in \mathbb{R}^{N_s \times N_o}, \quad S(z) \in \mathbb{R}^{N_s} \quad (9)$$

where $z = [z_1, z_2, \dots, z_{N_n}] \in \mathbb{R}^{N_n}$ in Ω_z is the input vector of RBFNN, N_s is neuron node number, N_o is output dimension of RBFNN, W is weight matrix, $S(z) = [s_1(z), s_2(z), \dots, s_{N_s}(z)]^T$ is hidden layer output function of RBFNN, and $s_i(z)$ is the i -th neuron output function, the Gaussian RBFNN function is chosen as follows

$$s_i(z) = e^{-\|z_i - c_{ij}\|/2b_i^2} \quad (10)$$

where $i = 1, 2, \dots, N_n$, $j = 1, 2, \dots, N_s$, c_{ij} is the center of the j -th neuron node for the i -th input signal, b_i is the width of the j -th neuron.

A number of research results have shown that for any continuous smooth function $\varphi(z) : \Omega_z \rightarrow R$ over a compact set $\Omega_z \subset R^{N_n}$ [32], [33], we can apply RBFNN (9) to approximate $\varphi(z)$. In particular, if N_s is chosen a sufficiently large value, such that the ideal bounded weight W^* exists, we have

$$\varphi(z) = W^{*T} S(z) + \mu(z) \quad (11)$$

where $\mu(z)$ is the approximation error, which is bounded as $|\mu(z)| < \mu^*$ with a given small constant μ^* .

RBFNN in (9) or in (11) has the following property, which will be used in the control design:

$$S(z)^T S(z) < N_s \quad (12)$$

Noting the ideal RBFNN weight W^* is unknown in practice, we often use \hat{W} as estimate weight of ideal weight W^* to approximate the unknown nonlinear function $\varphi(z)$. By designing an appropriate learning rule, the estimate \hat{W} can be renewed. Then, equation (11) can be rewritten as

$$\varphi(z) \approx \hat{W}^T S(z) \quad (13)$$

B. Desired Control

The ideal system tracking output is p_d^{k+2} . The dynamics of tracking error $e^{k+2} \in \mathbb{R}^n$ can be obtained as

$$e^{k+2} = p^{k+2} - p_d^{k+2} = L_p^k - L_G^k + M_\tau^k \tau^k + L_d^k - p_d^{k+2} \quad (14)$$

where p^{k+2} is defined in the (8).

There exists a continuous ideal control input τ_n^{*k} [29], such that

$$L_p^k - L_G^k + M_\tau^k \tau_n^{*k} - p_d^{k+2} = 0 \quad (15)$$

Lemma 2: There are positive constants $\underline{m}_\tau^* = 1/\bar{m}_\tau$ and $\bar{m}_\tau^* = 1/\underline{m}_\tau$, and M_τ^{k-1} is bounded as $\underline{m}_\tau^* \leq \|M_\tau^{k-1}\| \leq \bar{m}_\tau^*$. Thus, the predictor for two-step trajectory error e^{k+2} can be constrained as

$$\|e^{k+2}\| = \|L_d^k\| \leq \bar{\tau}_d^* \quad (16)$$

It is noted that the desired control τ_n^{*k} is not obtained with the unknown M_τ^{k-1} , L_p^k and L_G^k . We apply the adaptive RBFNN to learn and to approximate the desired input τ_n^{*k} , such that tracking error $e^{k+2} = 0$ after 2 steps can be achieved, if $\tau_d^k = 0$ and $\tau_d^{k-1} = 0$ in (14).

C. RBFNN Based Control

From Section IV-A, an ideal weight matrix W_τ^* exists, we apply RBFNN Gaussian function $S_\tau(\bar{z}^k)$ to approximate the ideal control input τ_n^{*k} as follows

$$\tau_n^*(\bar{z}^k) = W_\tau^{*T} S_\tau(\bar{z}^k) + \epsilon_\tau(\bar{z}^k) \quad (17)$$

where $S_\tau(\bar{z}^k) \in \mathbb{R}^{N_\tau}$ is the regression matrix, N_τ is neuron node number, $\|\epsilon_\tau(\bar{z}^k)\| \leq \epsilon_\tau^*$ with $\epsilon_\tau^* > 0$ is the approximation error, the ideal weight matrix $W_\tau^* \in \mathbb{R}^{N_\tau \times n}$, and the RBFNN input vector \bar{z} is designed as

$$\bar{z} = [p^{kT}, p^{k-1T}, v^{kT}, v^{k-1T}, \tau^{k-1T}, p_d^{k+2T}]^T \in \Omega_{\bar{z}}$$

where $\Omega_{\bar{z}}$ is a sufficient large compact set corresponding to Ω_{p_d} . It is easy to verify the ideal control τ_n^{*k} is bounded.

According to (14) and (15), we apply RBFNN to approximate the ideal control input $\tau_n^{*k}(\bar{z}^k)$, and introduce PD method to improve control performance, the system control input is designed as:

$$\begin{aligned} \tau^k &= -k_{pd} e^k + k_d e^{k-1} + \hat{\tau}_n(\bar{z}^k) \\ \hat{\tau}_n(\bar{z}^k) &= \hat{W}_\tau^T S_\tau(\bar{z}^k) \end{aligned} \quad (18)$$

where $k_{pd} = k_p + k_d > 0$, and $k_p > 0$, $k_d > 0$ are scaling factors, $\hat{W}_\tau^k \in \mathbb{R}^{N_\tau \times n}$ is used to approximate the ideal control input $\tau_n^{*k}(\bar{z}^k)$ in (17) with the compact set $\Omega_{\bar{z}}$.

According to equation (15), we have $p_d^{k+2} = L_p^k - L_G^k + M_\tau^k \tau_n^{*k}$. Then, equation (14) is rewritten as follows

$$e^{k+2} = L_p^k - p_d^{k+2} = M_\tau^k (\tau^k - \tau_n^{*k}) + L_d^k \quad (19)$$

For convenience, we define:

$$S_\tau^k = S_\tau(\bar{z}^k), \quad \epsilon_\tau^k = \epsilon_\tau(\bar{z}^k)$$

From Lemma 1, it is obvious that M_τ^k is bounded with \underline{m}_τ and \bar{m}_τ . Noting $\tilde{W}_\tau^k = \hat{W}_\tau^k - W_\tau^{*k}$, we substitute (17) and (18) into (19), then,

$$e^{k+2} = M_\tau^k (-k_{pd} e^k + k_d e^{k-1}) + M_\tau^k \tilde{W}_\tau^{kT} S_\tau^k + \tau_{dp}^k \quad (20)$$

where $\tau_{dp}^k = -M_\tau^k \epsilon_\tau^k + L_d^k$.

It is easy to show that $\|\tau_{dp}^k\| \leq \|M_\tau^k \epsilon_\tau^k\| + \|L_d^k\| \leq \bar{m}_\tau \epsilon_\tau^* + \bar{\tau}_d^* := \tau_{dp}^*$.

Then, the error equation in (20) can be converted as:

$$e^{k+2} + M_\tau^k k_{pd} e^k - M_\tau^k k_d e^{k-1} = M_\tau^k \tilde{W}_\tau^{kT} S_\tau^k + \tau_{dp}^k \quad (21)$$

We define a new error function as follows

$$e_1^{k+2} = e^{k+2} + M_\tau^k k_{pd} e^k - M_\tau^k k_d e^{k-1} \quad (22)$$

Substitute (20) into (22), the error function e_1^{k+2} is rewritten as

$$e_1^{k+2} = M_\tau^k \tilde{W}_\tau^{kT} S_\tau^k + \tau_{dp}^k \quad (23)$$

It is noted that the error function based on the adaptive RBFNN algorithms (23) is the $(k+2)$ th step error for robot system, then, we can obtain the k th step system error by defining $k_2 = k - 2$

$$e_1^k = M_\tau^{k_2} \tilde{W}_\tau^{k_2T} S_\tau^{k_2} + \tau_{dp}^{k_2} \quad (24)$$

where $\underline{m}_\tau \leq \|M_\tau^{k_2}\| \leq \bar{m}_\tau$ according to Lemma 1, and

$$e_1^k = e^k + M_\tau^{k_2} k_{pd} e^{k_2} - M_\tau^{k_2} k_d e^{k_2-1}$$

Based on system tracking error e_1^k , RBFNN update rule $\Delta \hat{W}_\tau^{k_2}$ for (18) is given by

$$\begin{aligned} \Delta \hat{W}_\tau^{k_2} &= -\Gamma_\tau S_\tau^{k_2} e_1^{kT} \\ \hat{W}_\tau^{k+1} &= \hat{W}_\tau^{k_2} + \Delta \hat{W}_\tau^{k_2} \end{aligned} \quad (25)$$

where $\Gamma_\tau = \gamma_\tau \mathbf{I}_{[N_\tau]} \in \Re^{N_\tau \times N_\tau}$ is a diagonal action system learning rate matrix with $\gamma_\tau > 0$.

D. Stability Analysis

It has been shown that an ideal control input $\tau_n^*(\bar{z}^k)$ exists and can guarantee $e^{k+2} = 0$, if the unknown disturbance $\tau_{dp}^k = 0$. Based on above all assumptions are only valid in compact set $\Omega_{\bar{z}}$, the system all outputs and inputs signal must be prove remain in corresponding compact sets.

A positive definite Lyapunov function V^k for the system (8) is chosen as

$$V^k = \sum_{j=0}^n \text{tr}[\tilde{W}_\tau^{k-2+jT} \Gamma_\tau^{-1} \tilde{W}_\tau^{k-2+j}] \quad (26)$$

where $\tilde{W}_\tau^k = \hat{W}_\tau^k - W_\tau^{*k}$.

Note the error function in (24), it is obvious that the Lyapunov

function V^k contains system tracking error, strategic signal error and parameter adaptation for RBFNN weights. The difference of (26) is given by

$$\Delta V^k = -e_1^{kT} \tilde{W}_\tau^{k_2T} S_\tau^{k_2} + b e_1^{kT} e_1^k \quad (27)$$

where $b = S_\tau^{k_2T} \Gamma_\tau S_\tau^{k_2}$.

Defining $A_w^{k_2} = M_\tau^{k_2} \tilde{W}_\tau^{k_2T} S_\tau^{k_2}$ and substituting (24) into (27), we have

$$\begin{aligned} \Delta V^k &= -2(A_w^{k_2} + \tau_{dp}^{k_2})^T M_\tau^{k_2-1} A_w^{k_2} \\ &\quad + b(A_w^{k_2} + \tau_{dp}^{k_2})^T (A_w^{k_2} + \tau_{dp}^{k_2}) \\ &\leq -(A_w^{k_2} + \tau_{dp}^{k_2})^T (M_\tau^{k_2-1} - b \mathbf{I}_{[n]}) \\ &\quad \times (A_w^{k_2} + \tau_{dp}^{k_2}) - A_w^{k_2T} M_\tau^{k_2-1} A_w^{k_2} \\ &\quad + \|J^k\|^2 \end{aligned} \quad (28)$$

where $\|J^k\|^2 = 2\tau_{dp}^{k_2T} M_\tau^{k_2-1} \tau_{dp}^{k_2} \leq \|J_\tau^k\|^2 = \frac{2\tau_{dp}^{*2}}{\underline{m}_\tau^*}$.

According to Lemma 2, it is easy to know that $M_\tau^{k_2-1}$ is symmetric positive definite matrix, and it can be bounded with $\underline{m}_\tau^* \leq \|M_\tau^{k_2-1}\| \leq \bar{m}_\tau^*$. If the eigenvalues of $M_\tau^{k_2-1}$ are $\lambda_i^{k_2}$, $i = 1, 2, \dots, n$, it is obvious that $\lambda_i^{k_2} > 0$.

We further define $\lambda_{max}^{k_2} = \max(\lambda_i^{k_2})$ and $\lambda_{min}^{k_2} = \min(\lambda_i^{k_2})$, then, $n\lambda_{min}^{k_2} \leq \|M_\tau^{k_2-1}\|^2 = \sum_{i=1}^n \lambda_i(M_\tau^{k_2-1T} M_\tau^{k_2-1}) \leq n\lambda_{max}^{k_2}$, $i = 1, 2, \dots, n$.

For convenience, we define $P_\tau^k = M_\tau^{k_2-1} - b \mathbf{I}_{[n]}$. The matrix P_τ^k being symmetric positive definite can be satisfied under following condition:

$$1 - b \frac{\bar{m}_\tau^*}{\sqrt{n}} > 0$$

Accord to the property in (12), it is obvious that $b = S_\tau^{k_2T} \Gamma_\tau S_\tau^{k_2} = \gamma_\tau S_\tau^{k_2T} S_\tau^{k_2} < \gamma_\tau N_\tau$. Analyse the difference of Lyapunov function in (28), the design parameter of controller are selected as

$$0 < \gamma_\tau < \frac{\sqrt{n}}{N_\tau \bar{m}_\tau^*} \quad (29)$$

Furthermore, the following theorem is presented to analyze stability of the system in (8), such that the closed-loop system stability and the trajectory tracking performance can be guaranteed by choosing appropriate parameters and adaptive weight gain of the controller.

Theorem 1: Assume that the conditions set above are satisfied, and define $B_w^{k_2} = A_w^{k_2} + \tau_{dp}^{k_2}$, then, we have

$$\Delta V^k \leq -B_w^{k_2T} P_\tau^k B_w^{k_2} + \|J_\tau^k\|^2 \quad (30)$$

Proof. There exists an invertible matrix Q_τ^k so that $P_\tau^k = Q_\tau^{kT} Q_\tau^k$. Accordingly, $\Delta V^k \leq 0$ can be satisfied under following conditions:

$$\|B_w^{k_2}\|^2 > \|J_\tau^k\| \|Q_\tau^k\|^{-1} \quad (31)$$

A discrete-time delay factor z^{-1} is introduced in (23), we have

$$e^k = (\mathbf{I}_{[n]} + M_\tau^k k_{pd} z^{k-2} - M_\tau^k k_d z^{k-3})^{-1} e_1^k \quad (32)$$

According to (31) and (32), we know there exists a finite running step K_τ , which makes $\|B_w^{k_2}\|^2 \leq \|J_\tau^k\| \|Q_\tau^k\|^{(-1)}$, then, $\|e^k\| \leq \|e_1^k\|$ under $(\mathbf{I}_{[n]} + M_\tau^k k_{pd} z^{k-2} - M_\tau^k k_d z^{k-3})^{-1}$ being Hurwitz-stable for all $k > K_\tau$. Consider the boundedness of M_τ^k and τ_{dp}^* , $B_w^{k_2} = A_w^{k_2} + \tau_{dp}^{k_2}$, such that the error e_1^k is bounded as

$$\begin{aligned} \|e_1^k\|^2 &= e_1^{k^T} e_1^k \leq 2A_w^{k_2^T} A_w^{k_2} + 2\tau_{dp}^{k_2^T} \tau_{dp}^{k_2} \\ &< 4\|J_\tau^k\| \|Q_\tau^k\|^{(-1)} + 6\tau_{dp}^{*2} \end{aligned} \quad (33)$$

or, we can get

$$\|e^k\| \leq \|e_1^k\| < \sqrt{4\|J_\tau^k\| \|Q_\tau^k\|^{(-1)} + 6\tau_{dp}^{*2}} \quad (34)$$

the proof is complete.

V. SIMULATION STUDIES

To verify the above developed adaptive RBFNN control approach, a testing example, 2-DOF robot manipulator interacting is used in this section.

A. Robot Manipulator Dynamics Model

The following parameters of robot manipulator are specified.

The mass are $m_1 = m_2 = 1.0\text{kg}$, the length are $l_1 = l_2 = 0.2\text{m}$, the inertia are $I_1 = I_2 = 0.003\text{kgm}^2$, the distance are $l_{c1} = l_{c2} = 0.1\text{m}$.

Then, dynamics of the robot manipulator with $G(q) = \mathbf{0}_{[2]}$ is given as

$$\begin{aligned} M(q) &= [M_{11} \ M_{12}; M_{21} \ M_{22}] \\ C(q) &= [C_{11} \ C_{12}; C_{21} \ C_{22}] \end{aligned} \quad (35)$$

where

$$\begin{aligned} M_{11} &= m_1 l_{c1}^2 + m_2 (l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos(q_2)) + I_1 + I_2 \\ M_{12} &= M_{21} = m_2 (l_{c2}^2 + l_1 l_{c2} \cos(q_2)) + I_2 \\ M_{22} &= m_2 l_{c2}^2 + I_2 \\ C_{11} &= -m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2 \\ C_{12} &= -m_2 l_1 l_{c2} \sin(q_2) (\dot{q}_1 + \dot{q}_2) \\ C_{21} &= m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1, \quad C_{22} = 0 \end{aligned}$$

The external force torque may be caused by disturbance, a smaller and a larger amplitude force torque τ_{ds} and τ_{db} are assumed as, respectively,

$$\begin{aligned} \tau_{ds} &= [0.05 \cos(0.01t) \cos(q_1), 0.05 \cos(0.01t) \cos(q_2)]^T \\ \tau_{db} &= [40 \cos(0.01t) \cos(q_1), 40 \cos(0.01t) \cos(q_2)]^T \end{aligned}$$

Two different types of desired trajectory q_{dd} and q_{dg} are assumed as

$$\begin{aligned} q_{dd} &= [q_{dd1}, q_{dd2}]^T = \begin{bmatrix} 1.5 + 0.5(\sin(0.3t) + \sin(0.2t)) \\ 1.5 + 0.5(\cos(0.4t) + \sin(0.3t)) \end{bmatrix} \\ q_{dg} &= [q_{dg1}, q_{dg2}]^T = \begin{bmatrix} 0.6 \sin(\cos(\pi t/200)) + 0.4 \\ 0.5 \sin(\sin(\pi t/200)) \end{bmatrix} \end{aligned}$$

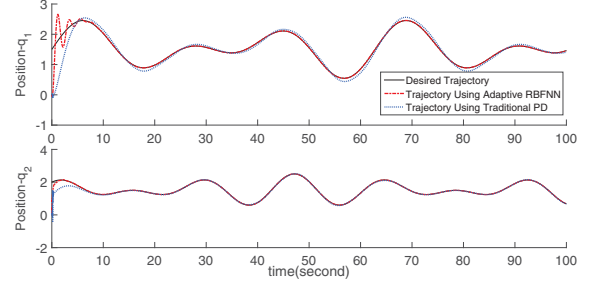


Fig. 1. Position trajectory for tracking q_{dd} using adaptive RBFNN and PD for a small disturbance τ_{ds}

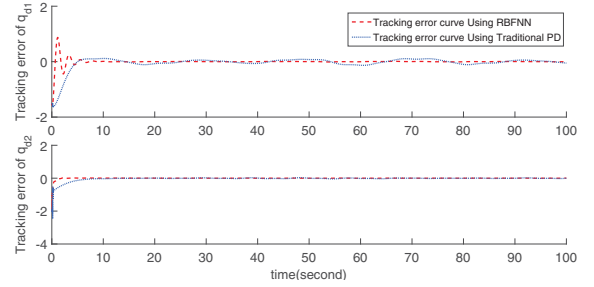


Fig. 2. Error curve for tracking ideal q_{dd} using adaptive RBFNN and PD for a small disturbance τ_{ds}

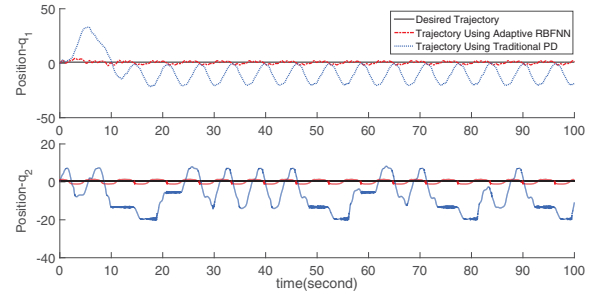


Fig. 3. Position trajectory for tracking q_{dg} using adaptive RBFNN and PD for a large disturbance τ_{db}

B. Test Results

The initial states of robot manipulator in (35) are $q(0) = [0, 0]^T$ and $\dot{q}(0) = [0, 0]^T$. We construct the adaptive RBFNN $\hat{W}_\tau^{k^T} S_\tau^k$, which approximates system tracking error using $N_\tau = 4096$ with all the centres of Gaussian function evenly in $[-1; 1]$ and all the widths $b = 1$. The design parameters are chosen as $\gamma_\tau = 0.01$, $k_p = 6.5$, $k_d = 115$. The initial weights $\hat{W}_\tau(0) = \mathbf{0}_{[2 \times N_\tau]}$. Simulation results are presented with the controller sampling interval $T = 0.01\text{s}$.

To show the effectiveness, we use the above same design parameters, and compare the position trajectory accuracy and capability between the adaptive RBFNN control and traditional PD control $\tau^k = -k_p e^k - k_d(e^k - e^{k-1})$ for the robot manipulator (4) with τ_{ds} and τ_{db} in Figs. 1-3.

Fig.1-2 show trajectory tracking trajectories and error trajectories of q_1 and q_2 for the desired q_{dd} with added

external disturbance torque τ_{ds} , respectively. Fig.3 shows trajectory tracking trajectories of q_1 and q_2 for the desirable q_{dg} with added external disturbance torque τ_{db} .

Comparing with a traditional PD control based on the above simulation results with a small disturbance signal τ_{ds} , the first joint of the proposed discrete-time adaptive RBFNN control has an initial error and deviates from the desired trajectory for less than 8s, but it can adjust itself quickly to achieve the desired trajectory; and the second joint using the proposed control has also an excellent tracking performance than the traditional PD control. Furthermore, a large disturbance signal τ_{db} added to test tracking performance of the proposed control for the desired trajectory q_{dg} , the simulation results are given in Fig.3, which shows that two joints of robot manipulator have achieved satisfied tracking trajectories using adaptive RBFNN control.

VI. CONCLUSION

An discrete-time adaptive RBFNN has been developed for a class of uncertain robot manipulators to achieve precise tracking control performance. The adaptive RBFNN controller is designed to estimate system error, where the control law is adaptively tuned online. Two different types of given trajectory and two kinds of external disturbances are used to test the performance of the proposed approach in the simulation. The proposed discrete-time adaptive RBFNN control is able to overcome effects of external disturbances and internal uncertainties. Not only the closed-loop system stability is guaranteed via Lyapunov stability analysis, but also excellent tracking performance is achieved.

REFERENCES

- [1] C. Pukdeboon. Lyapunov Optimizing Sliding Mode Control for Robot Manipulators. *Applied Mathematical Sciences*, 7(63), 3123-3139, 2013.
- [2] F. L. Lewis, D. M. Dawson, C. T. Abdallah. *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. 2nd. Marcel Dekker, Inc., New York, USA, 2004.
- [3] F. L. Lewis, S. Jagannathan, A. Yesildirak. *Neural Network Control of Robot Manipulators and Non-linear Systems*. Taylor Francis: London, UK:1 Gunpowder Square, EC4A 3DE, 1999.
- [4] X. Ren, A. B. Rad, F. L. Lewis. Neural Network-based Compensation Control of Robot Manipulators with Unknown Dynamics. *Proc. of the 2007 American control conference*, New York, USA, pp.13-18, 2007.
- [5] G. Palli, C. Melchiorri, A. D. Luca. On the Feedback Linearization of Robots with Variable Joint Stiffness. *IEEE International Conference, Robotics and Automation*, Pasadena, California, USA, pp.19-23, 2008.
- [6] N. S. Bedrossian, M. W. Spong. Feedback Linearization of Robot Manipulators and Riemannian Curvature. *Journal of Robotic Systems*, 12(8), 541-552, 1995.
- [7] J. Yang, J. H. Kim. Sliding Mode Control for Trajectory Tracking of Nonholonomic Wheeled Mobile Robots. *Robotics and Automation*, *IEEE Trans.* 15(3), 3123-3139, 1999.
- [8] H. Guo, Y. Liu, G. Liu, H. Li. Cascade Control of a Hydraulically Driven 6-dof Parallel Robot Manipulator Based on a Sliding Mode. *Control Engineering Practice*, 16(9), 1055-1068, 2008.
- [9] Y. Pi, X. Wang. Trajectory Tracking Control of a 6-dof Hydraulic Parallel Robot Manipulator with Uncertain Load Disturbances. *Control Engineering*, 19(2)185-193, 2011.
- [10] M. N. Mahyuddin, G. Herrmann, S. G. Khan. A Novel Adaptive Control Algorithm in Application to a Humanoid Robot Arm. *Advances in Autonomous Robotics*, 7429, 25-36, 2012.
- [11] T. F. Pazelli, M. H. Terra, A. A. Siqueira. Experimental Investigation on Adaptive Robust Controller Designs Applied to a Free-floating Space Manipulator. *Control Engineering Practice*, 19(4), 395-408, 2011.
- [12] N. Mendes, P. Neto. Indirect Adaptive Fuzzy Control for Industrial Robots: A Solution for Contact Applications. *Expert Systems With Applications*, 42(22), 8929-8935, 2015.
- [13] J. S. Park, G. S. Han, H. S. Ahn, D. H. Ki. Adaptive Approaches on the Sliding Mode Control of Robot Manipulators. *Control, Automation and Systems Engineering, Trans.*, 3(1), 15-20, 2001.
- [14] A. B. B. Sharkawy, S. A. Salman. An Adaptive Fuzzy Sliding Mode Control Scheme for Robotic Systems. *Intelligent Control and Automation*, 2, 299-309, 2011.
- [15] X. Li, C. C. Cheah. Adaptive Neural Network Control of Robot Based on a Unified Objective Bound. *Control Systems Technology, IEEE Trans.*, 22(3), 1032-1043, 2014.
- [16] C. L. P. Chen, G. Wen, Y. Liu, F. Wang. Adaptive Consensus Control for a Class of Nonlinear Multiagent Time-delay Systems Using Neural Networks. *Neural Networks and Learning Systems, IEEE Trans.*, 25(6), 1217-1226, 2014.
- [17] J. Liu, Y. Lu. Adaptive RBF Neural Network Control of Robot with Actuator Nonlinearities. *Journal of Control Theory and Applications*, 8(12), 249-256, 2010.
- [18] S. Jagannathan, F. L. Lewis. Multilayer Discrete-time Neural-net Controller with Guaranteed Performance. *Neural Networks, IEEE Trans.*, 7(1), 107-130, 1996.
- [19] M. M. Fateh, S. M. Ahmadi, S. Khorashadizadeh. Adaptive RBF Network Control for Robot Manipulators. *Journal of AI and Data Mining*, 2(2), 159-166, 2014.
- [20] W. Zhang, X. Ye, L. Jiang, Y. Fang. Robust Control for Robotic Manipulators Base on Adaptive Neural Network. *The Open Mechanical Engineering Journal*, 8, 497-502, 2014.
- [21] S. G. Tzafestas. Dynamic Modelling and Adaptive Control of Industrial Robots: The state-of-art. *Systems Analysis Modelling Simulation*, 1, 243-266, 1989.
- [22] L. Zhai, T. Chai, S. S. Ge, T. H. Lee. Stable Adaptive Neural Network Control of MIMO Nonafne Nonlinear Discrete-Time Systems. *Decision and Control*, 47th IEEE Conference, Cancun, Mexico, 9-11, 2009.
- [23] F. Sun, L. Li, H. Li, H. Liu. Neuro-Fuzzy Dynamic-Inversion-Based Adaptive Control for Robotic Manipulators Discrete Time Case. *Industrial Electronics, IEEE Trans.*, 54(3), 1342-1351, 2007.
- [24] L. Zhai, C. Yang, S. S. Ge, T. Chai, T. H. Lee. Direct Adaptive NN Control of MIMO Nonlinear Discrete-Time Systems using Discrete Nussbaum Gain. *The International Federation of Automatic Control*, 17th World Congress, Seoul, Korea, 6508-6512, 2008.
- [25] S. Dai, C. Wang, F. Luo. Identification and Learning Control of Ocean Surface Ship Using Neural Networks. *Industrial Informatics, IEEE Trans.*, 8(4), 801-809, 2012.
- [26] F. Sun, H. Li, L. Li. Robot Discrete Adaptive Control Based on Dynamic-inversion Using Dynamical Neural Networks. *Automatica*, 38(11), 1977-1983, 2002.
- [27] A. Zagoranos, S. G. Tzafestas, G. S. Stavrakakis. On line Discrete Time Control of Industrial Robots. *Robotics and Autonomous Systems*, 14(4), 289-299, 1995.
- [28] Y. Li, C. Yang, S. S. Ge, T. H. Lee. Adaptive Output Feedback NN Control of a Class of Discrete-Time MIMO Nonlinear Systems With Unknown Control Directions. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Trans.*, 41(2), 5007-517, 2011.
- [29] C. Yang, S. S. Ge, C. Xiang, T. Chai, T. H. Lee. Output Feedback NN Control for Two Classes of Discrete-time Systems with Unknown Control Directions in a Unified Approach. *Neural Networks, IEEE Trans.*, 19(11), 1983-1886, 2008.
- [30] C. Yang, S. S. Ge, T. H. Lee. Output Feedback Adaptive Control of a Class of Nonlinear Discrete-time Systems with Unknown Control Directions. *Automatica*, 45(1), 270-276, 2009.
- [31] S. S. Ge, C. Yang, T. H. Lee. Adaptive Predictive Control Using Neural Network for a Class of Pure-Feedback Systems in Discrete Time. *Neural Networks, IEEE Trans.*, 19(9), 1599-1613, 2008.
- [32] A. Smith, C. Yang, H. Ma, P. Culverhouse, A. Cangelosi, E. Burdet. Novel Hybrid Adaptive Controller for Manipulation in Complex Perturbation Environments. *PLOS ONE*, 10(6), 1-19, 2015.
- [33] C. Yang, Z. Li, R. Cui, B. Xu. Neural Network-based Motion Control of An Under-actuated Wheeled Inverted Pendulum Model, *IEEE Transactions on Neural Networks and Learning Systems*, 25(11), 2004-2016, 2014.